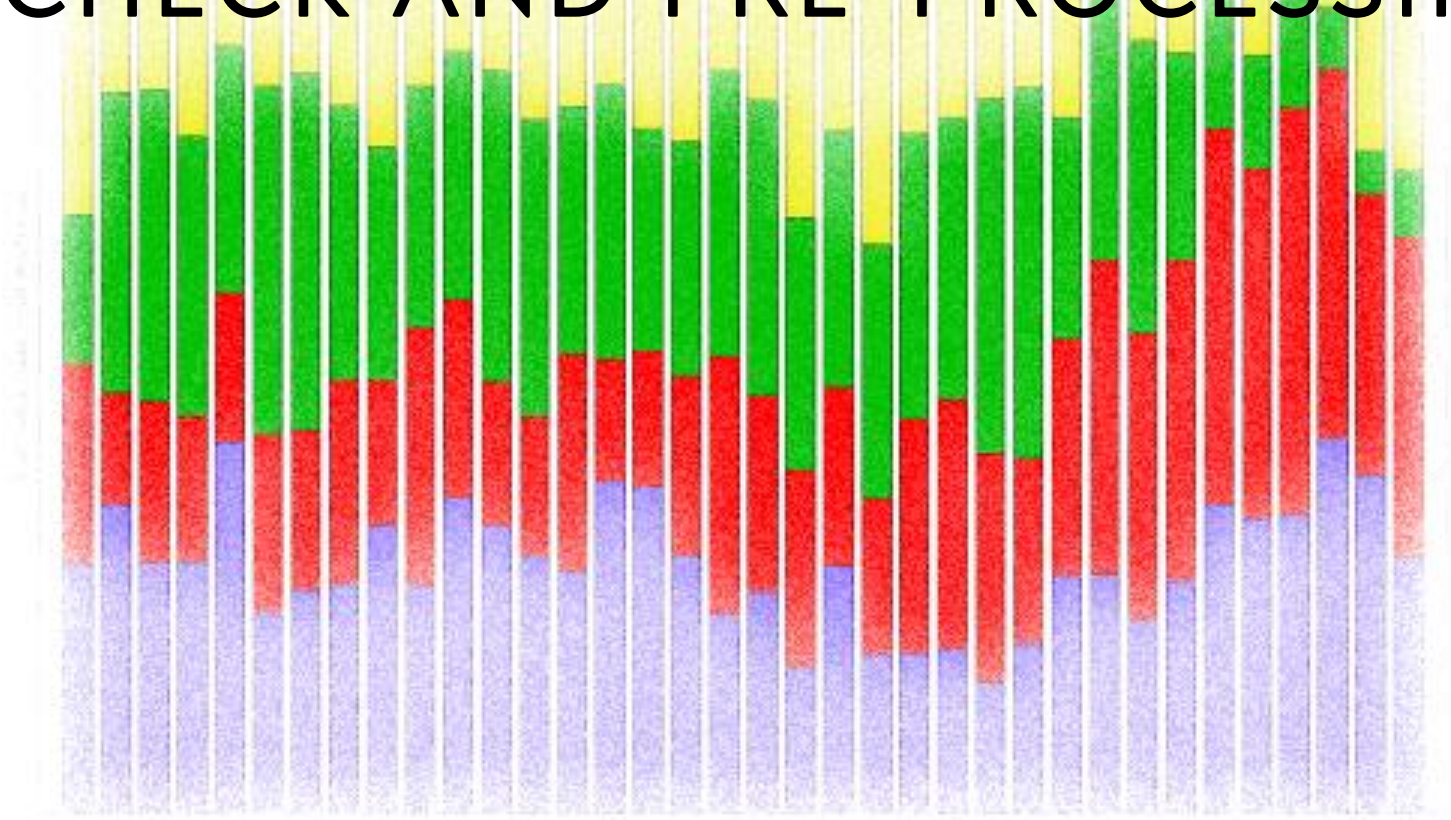
SEQUENCE QUALITY MEASURES, QUALITY CHECK AND PRE-PROCESSING



104-SHRUTI BHUJBAL
106-SHUBHANGI DETHE
116-KAJAL LAMANE
117-RITIKA MAURYA

SEQUENCE QUALITY MEASURES- PHRED QUALITY SCORE.

What is sequence quality measures?

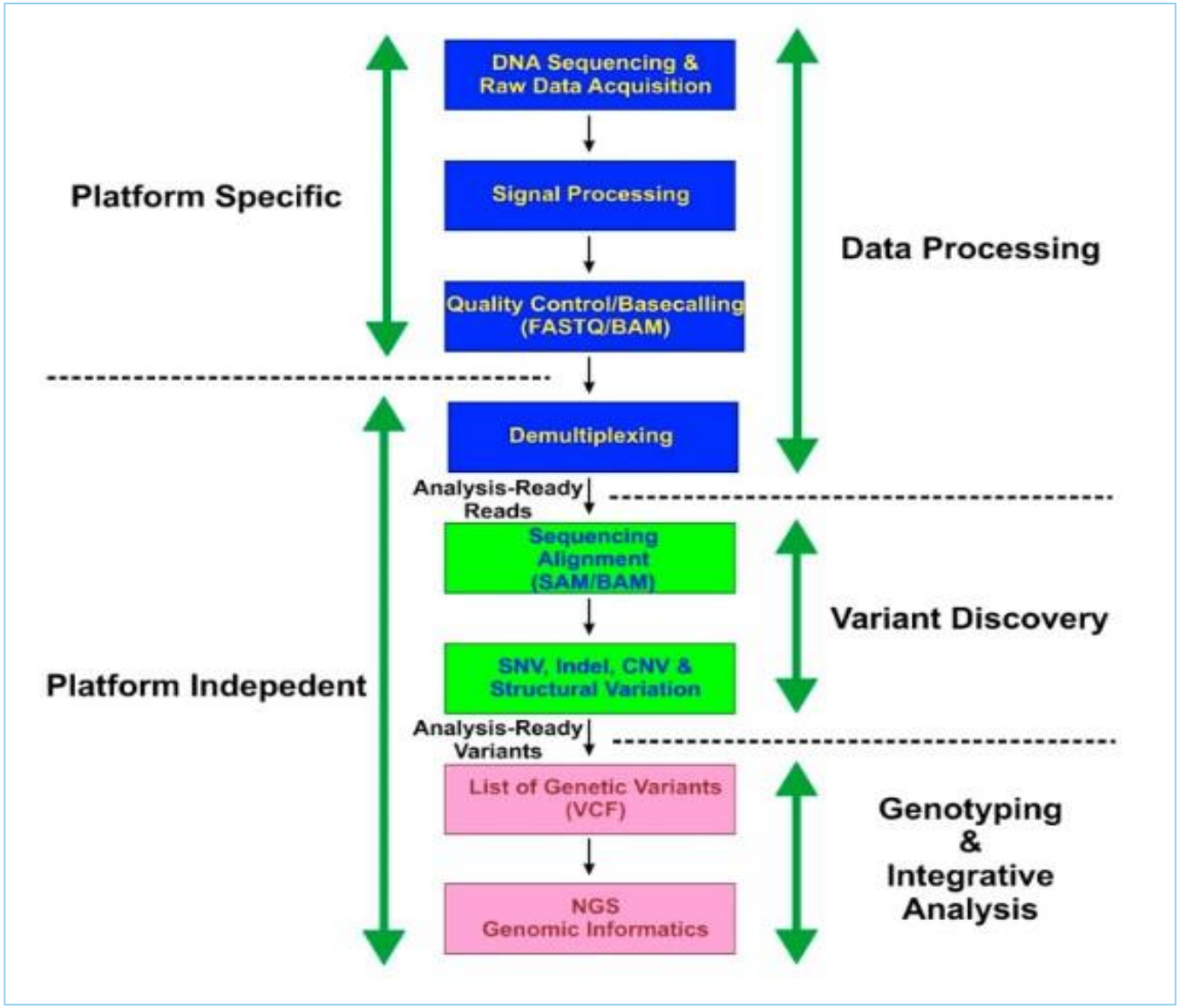
- Sequencing quality scores measure the probability that a base is called incorrectly.
- Most sequencers will generate a quality control report as part of their analysis pipeline, but this is usually only focused on identifying problems which were generated by the sequencer itself.
- There are many sequences quality controls tools such as FastQC, FastX, Sickle, and RNA-SeQC. FastQC can visually view the quality of the segment. FastX can remove the low quality of the callor read segment. For double-ended sequences, Sickle can simultaneously remove the corresponding reverse read segment while filtering out the forward - segment of a lot of low-quality base, and vice versa.
- RNA-SeQC calculation of RNA-seq data quality indicators used to guide experimental design, quality control and optimization analysis, such as sequences depth (depth of coverage), the alignment area (intron, exon, gene region), rDNA content and so on. RNA-SeQC can also be the length of the sequences alignment of the resuts of statistical analysis, to get a number of quality control indicators.

What is Phred Quality Score?

PHRED - Phil's Read Editor (University of Washington Genome Center)
Phred quality score is used to indicate the measure of base quality in DNA sequencing. High consistency of a sequenced base is indicated by greater values of Phred. A Phred Score of 20 indicates the likelihood of finding 1 incorrect base call among 100 bases. In other words, the precision of the base call is 99%.

How do you read a Phred quality score?
A higher score indicates a higher probability that a particular decision is correct, while conversely, a lower score indicates a higher probability that the decision is incorrect. The Phred quality score (Q) is logarithmically related to the error probability (E).

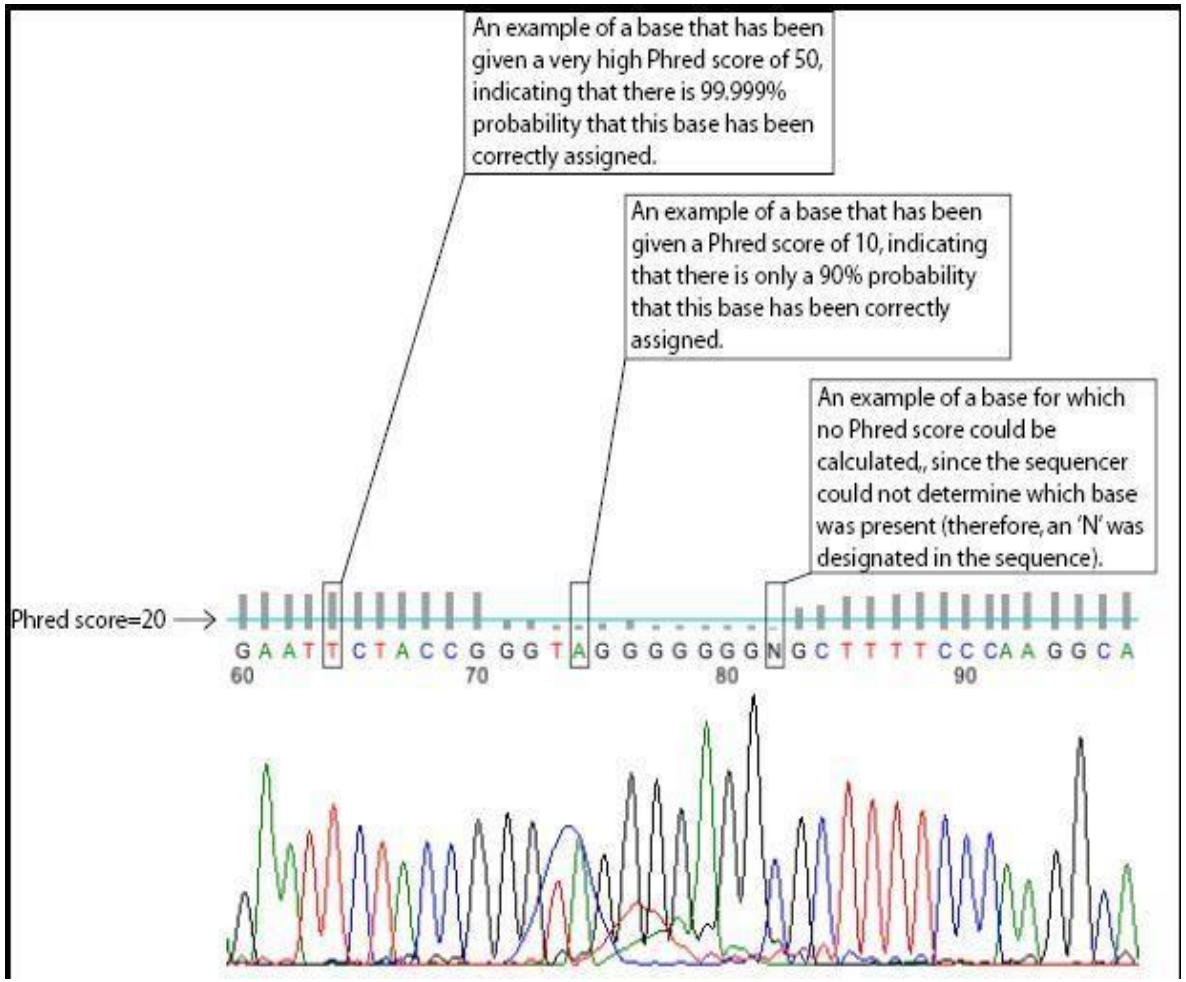
Why phred quality score use?
To indicate the measure of base quality in DNA sequencing. High consistency of a sequenced base is indicated by greater values of Phred. A Phred Score of 20 indicates the likelihood of finding 1 incorrect base call among 100 bases. In other words, the precision of the base call is 99%.


DNA Sequencing & Raw Data Acquisition
Signal Processing
Quality Control/Basecalling (FASTQ/BAM)
Demultiplexing
Analysis-Ready Reads
Sequencing Alignment (SAM/BAM)
SNV, Indel, CNV & Structural Variation
Analysis-Ready Variants
List of Genetic Variants (VCF)
NGS Genomic Informatics
Platform Specific
Platform Indepedent
Data Processing
Variant Discovery
Genotyping & Integrative Analysis

Phred-quality scores

- Originally developed in phred program (a base calling software)
- Developed by Ewing et all (1998)
- To be used in HGP (Human genome Project).
- The property that is logarithmically related with base call error probability.
- High accuracy of phred quality Scores makes them Standard for measuring sequencing quality.
- P = Probability that a base is wrong
- **Formula : Q =** $-10\log_{10} P$
- **P =** $10^{-Q/10}$
- Q is known as phred quality score
- Also written as Qsanger
- What is the value of P (Probability that a base is wrong) for Q = 40
- $P=10^{-Q/10}$
- $P=10^{-40/10}$
- $P=10^{-4}$
- P=0.0001

Phred Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%
60	1 in 1000,000	99.9999%



QUALITY CHECK

What is FastQC ?

- Aims to provide a simple way to do some quality control checks on raw sequence data coming from high throughput sequencing pipelines.

- Provide a QC report which can spot problems which originate either in the sequencer or in the starting library material.

The main functions of FastQC are

- Import of data from BAM, SAM or FastQ files (any variant)
- Providing a quick overview to tell you in which areas there may be problems
- Summary graphs and tables to quickly assess your data
- Export of results to an HTML based permanent report

FastQC supported files formats

- FastQ (all quality encoding variants)
- Colorspace FastQ
- GZip compressed FastQ
- SAM
- BAM

BASIC STATISTICS

FastQC Report

Summary

- Basic Statistics
- Per base sequence quality
- Per tile sequence quality
- Per sequence quality scores
- Per base sequence content
- Per sequence GC content
- Per base N content
- Sequence Length Distribution
- Sequence Duplication Levels
- Overrepresented sequences
- Adapter Content

Basic Statistics

Measure	Value
Filename	sample-fastqc.gz
File type	Conventional base calls
Encoding	Sanger / Illumina 1.9
Total Sequences	10602766
Total Bases	1 Gbp
Sequences flagged as poor quality	0
Sequence length	101
%GC	49

Per base sequence quality

PER BASE SEQUENCE QUALITY

BoxWhisker type plot

- The central red line - median value
- The yellow box - the inter-quartile range (25-75%)
- The upper and lower whiskers represent the 10% and 90% points
- The blue line represents the mean quality

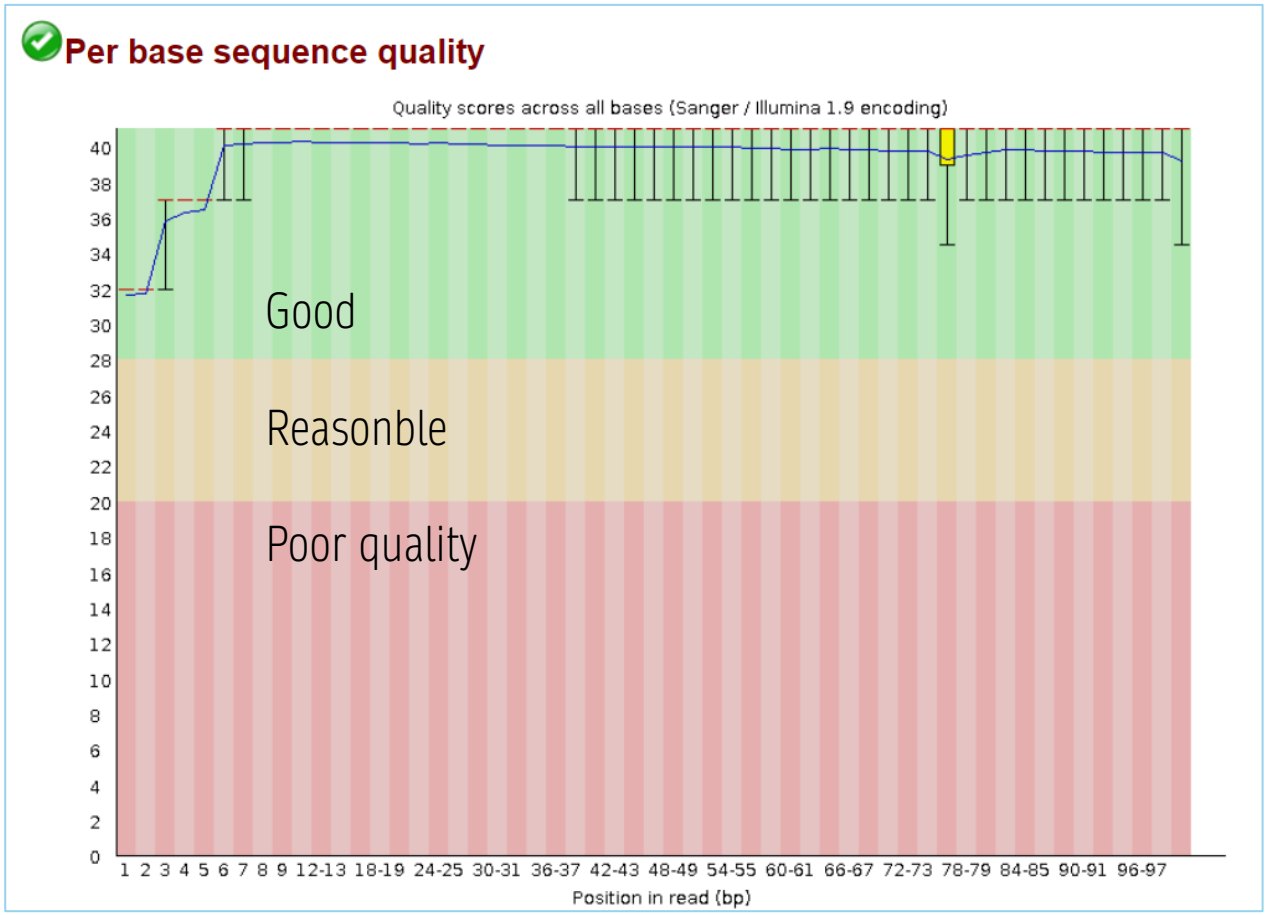
Warning

lower quartile - less than 10

median for any base is less than 25.

Failure

lower quartile - less than 5 median for any base is less than 20.



PER TILE SEQUENCE QUALITY

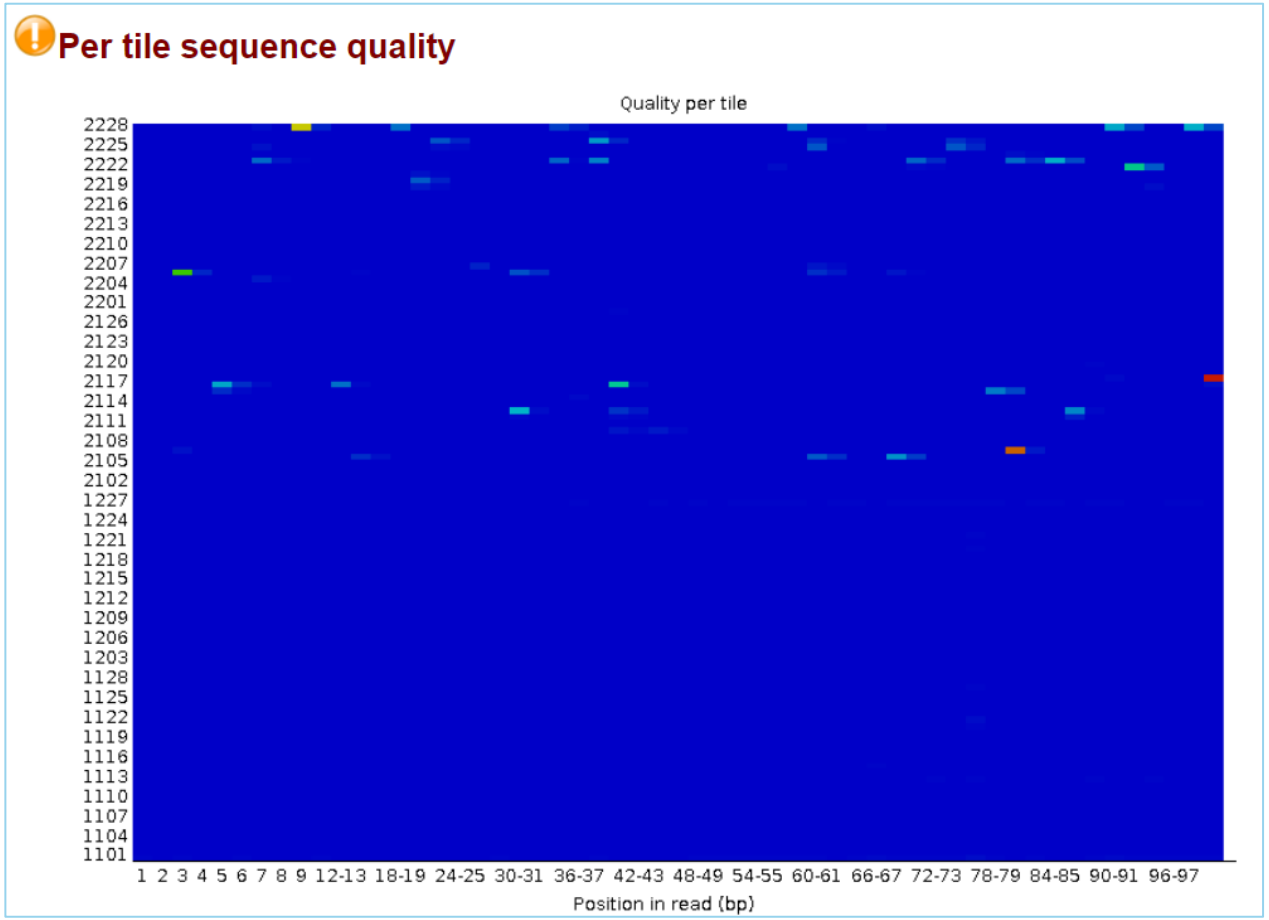
- The plot shows the deviation from the average quality for each tile
- To see loss in quality associated with only one part of the flowcell (from which each read came).

Warning

Mean Phred score > 2 and less than the mean (base across all tiles).

Failure

Mean Phred score > 5 and less than the mean (base across all tiles).



PER SEQUENCE QUALITY SCORE

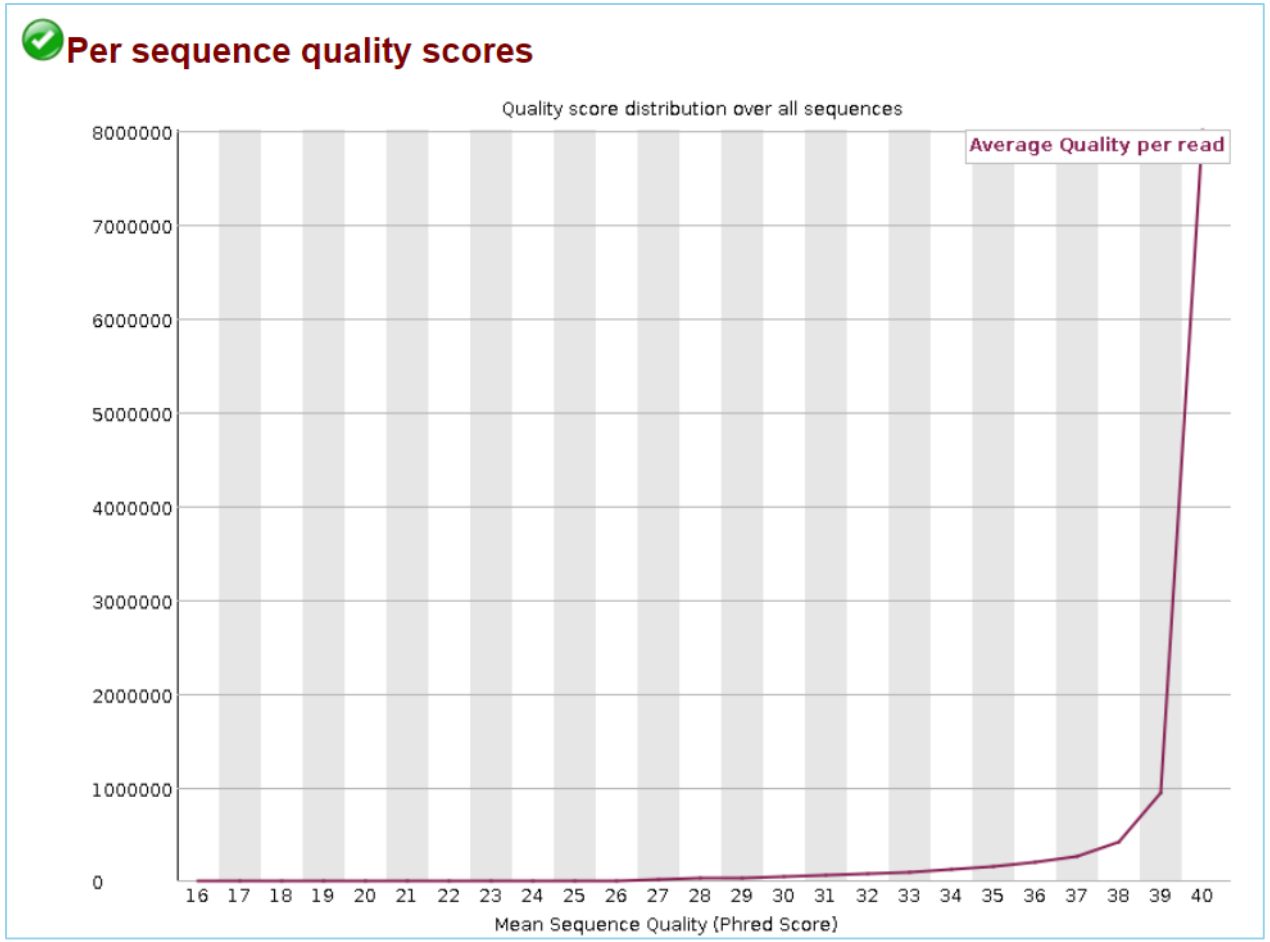
- To see if subset of your sequences have universally low quality values.
- Errors here usually indicate a general loss of quality within a run.

Warning

Mean quality < 27 - this equates to a 0.2% error rate.

Failure

Mean quality < 20 - this equates to a 1% error rate.



PER BASE SEQUENCE CONTENT

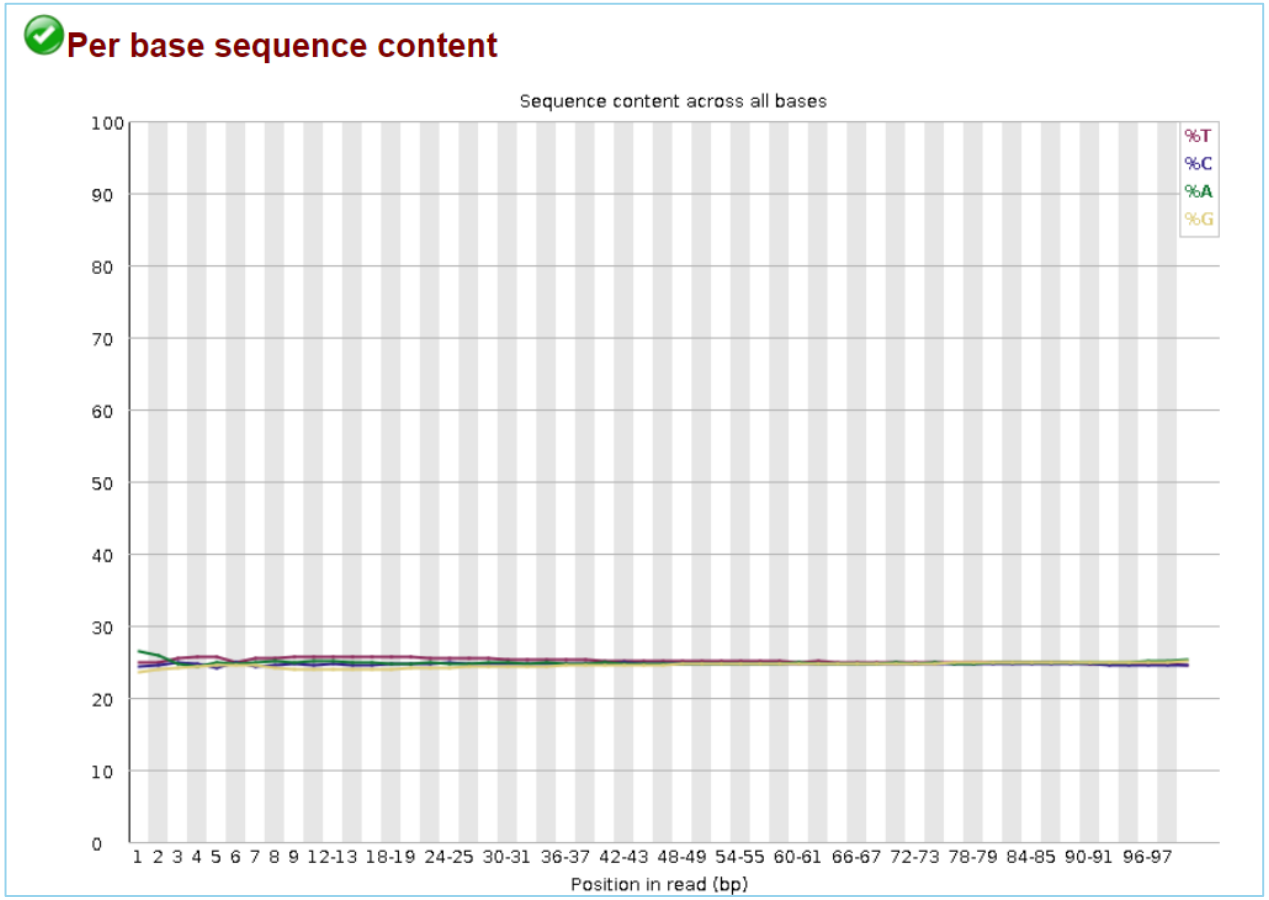
- Proportion of each base position in a sequence.
- No difference between the different bases of a sequence run, shows the lines in this plot should run parallel with each other

Warning

Difference between A and T, or G and C is greater than 10% in any position.

Failure

Difference between A and T, or G and C is greater than 20% in any position.



PER SEQUENCE GC CONTENT

- Measures the GC content across the whole length of each sequence

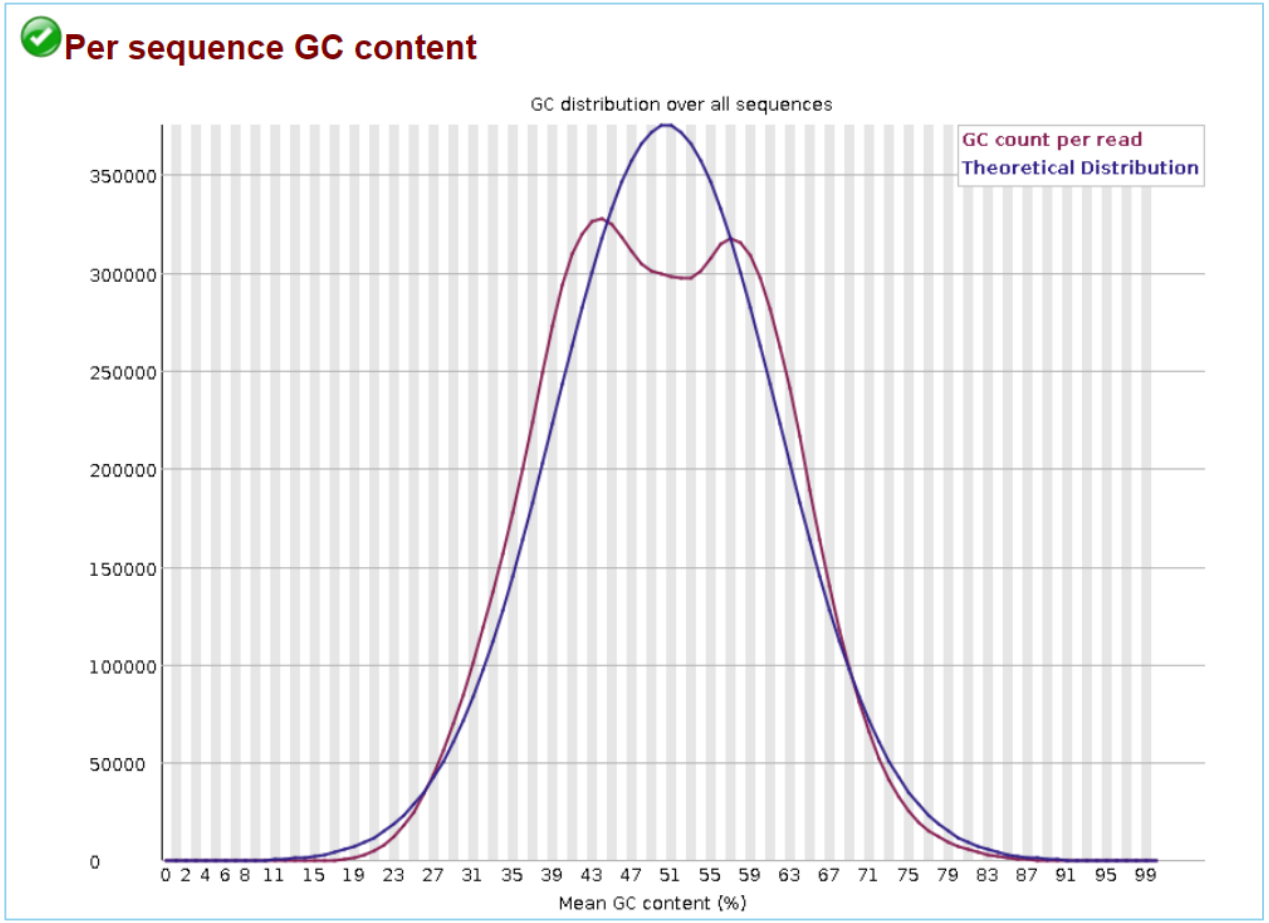
- the central peak corresponds to the overall GC content

Warning

Sum of the deviations from the normal distribution > 15% of the reads.

Failure

Sum of the deviations from the normal distribution > 30% of the reads.



PER BASE N CONTENT

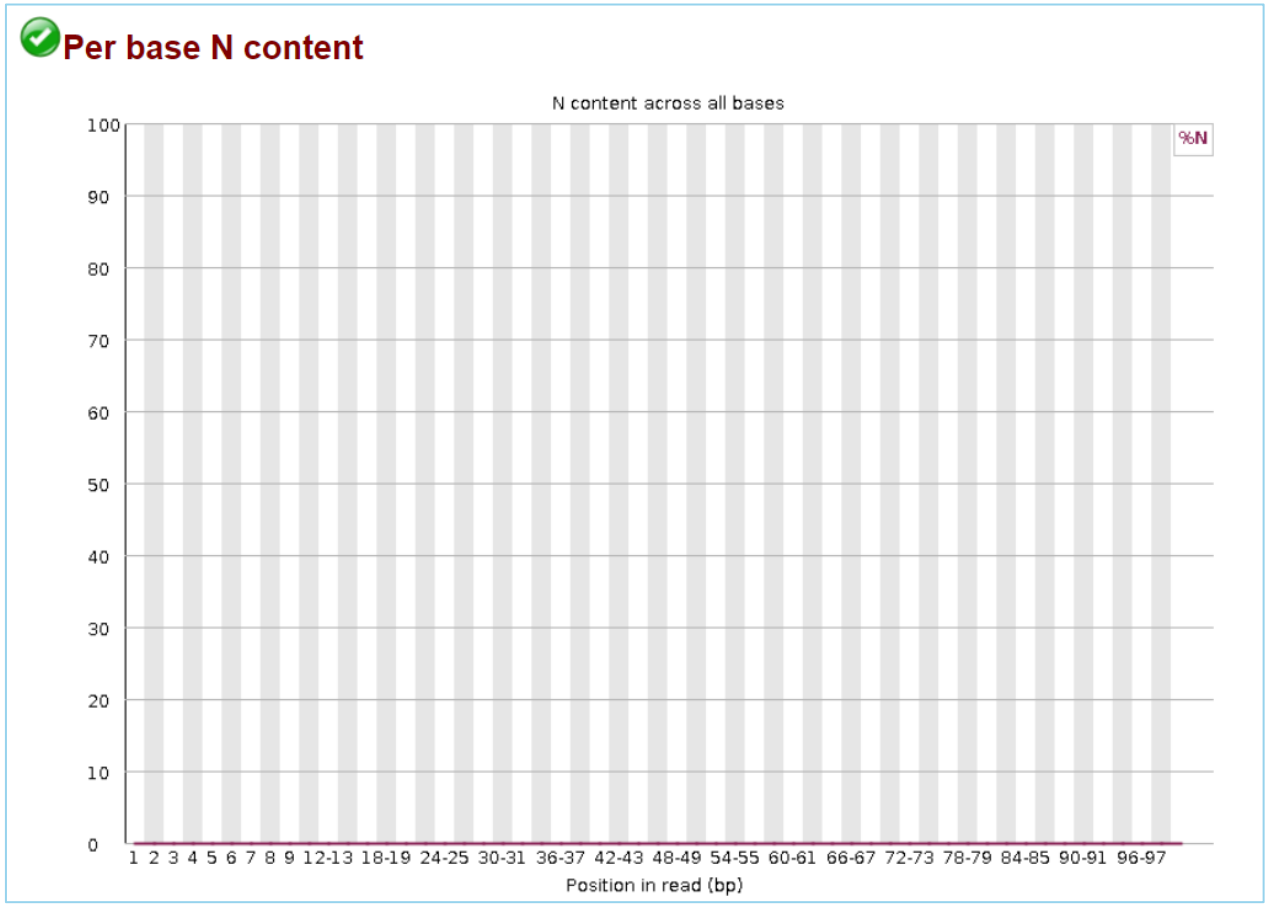
- Measures the amount of N called at each position in a sequence

Warning

Position shows an N content of >5%.

Failure

Position shows an N content of >20%.



SEQUENCE LENGTH DISTRIBUTION

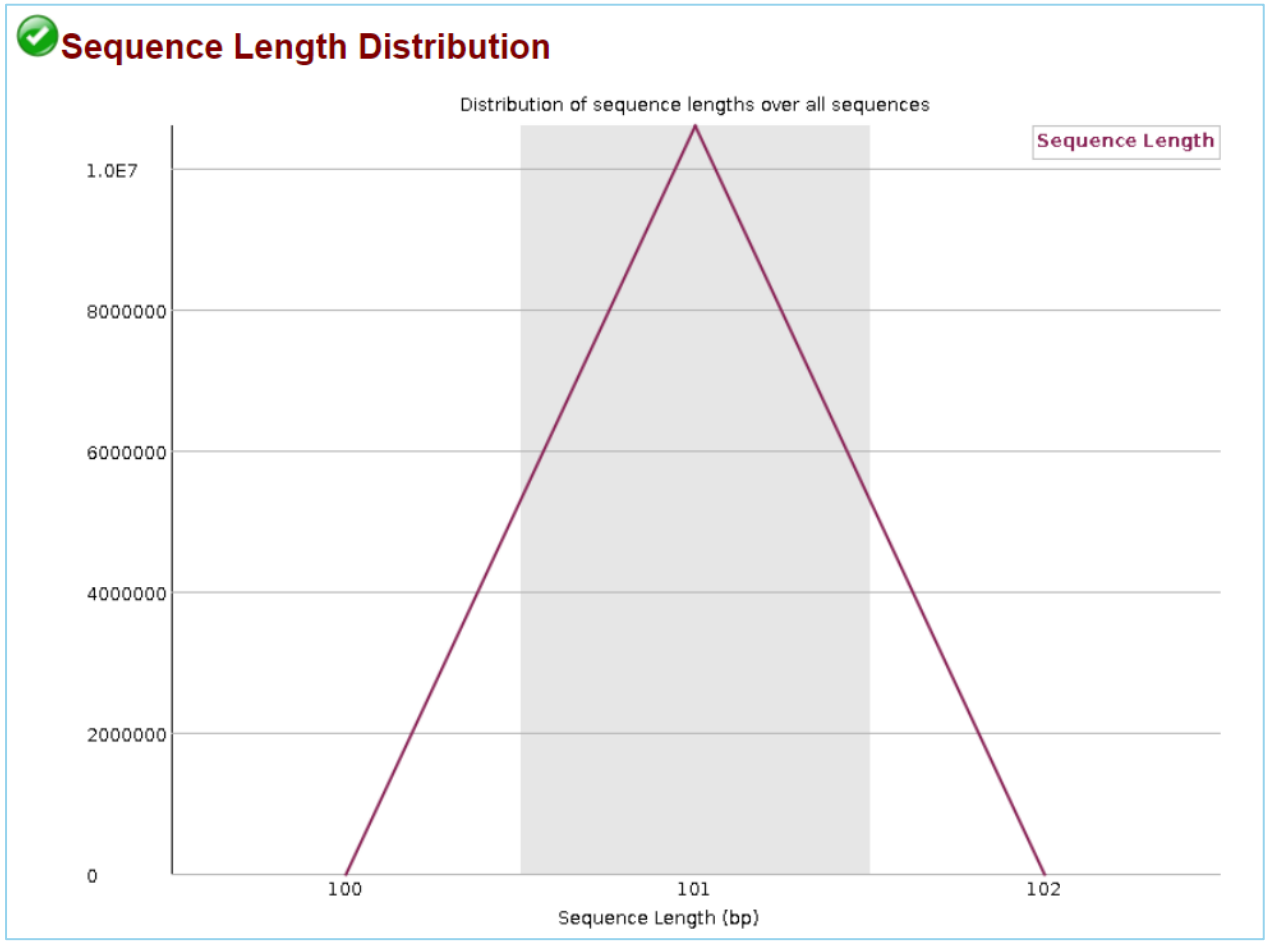
- Shows the distribution of fragment sizes in the sequence which was analysed.

Warning

if all sequences are not the same length.

Failure

if any of the sequences have zero length.



SEQUENCE DUPLICATION LEVEL

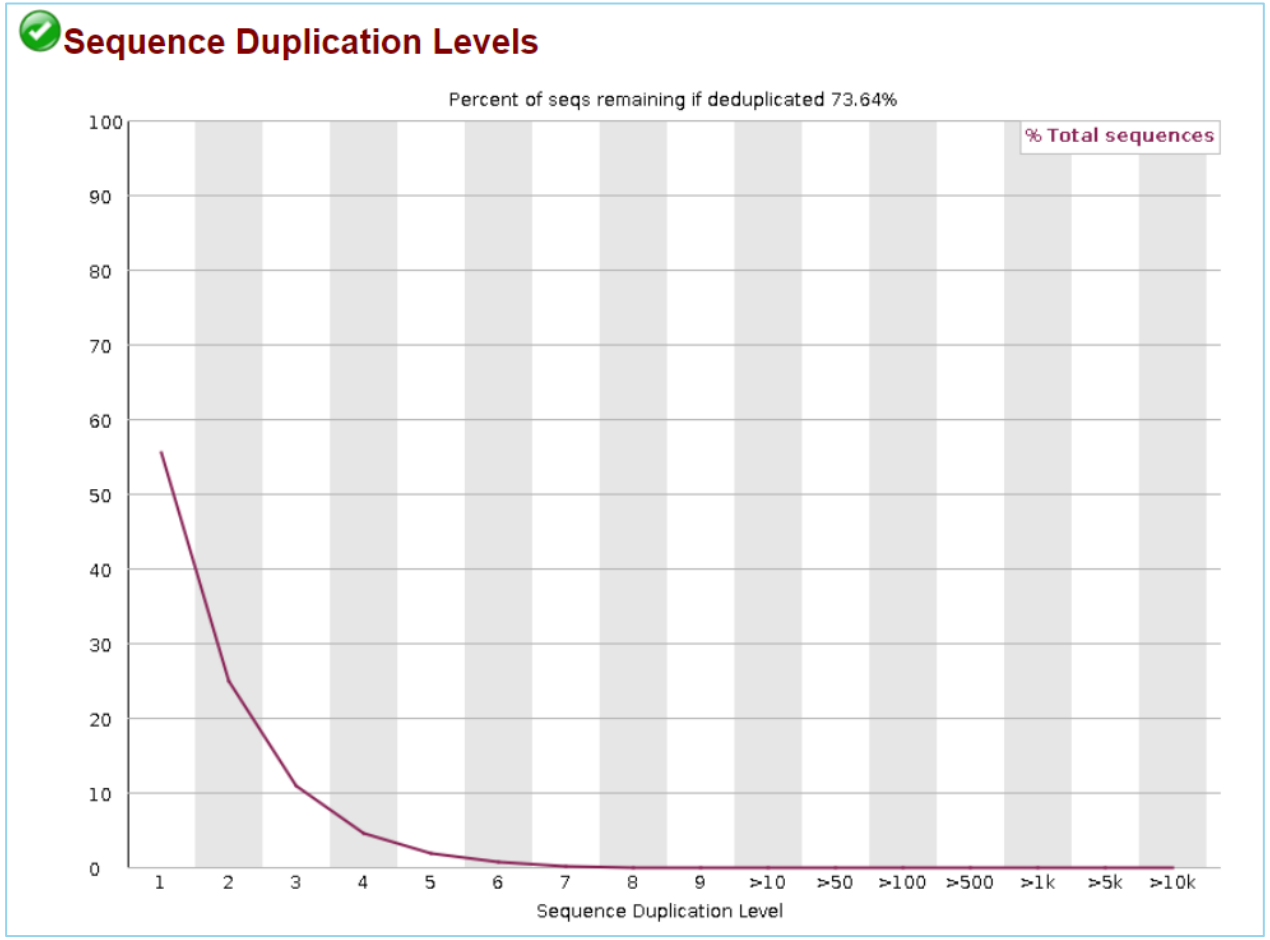
- Counts the degree of duplication for every sequence

Warning

if non-unique sequences make up more than 20% of the total.

Failure

if non-unique sequences make up more than 50% of the total.



OVERREPRESNTED SEQUENCES

- Lists all of the sequence which make up more than 0.1% of the total
- sequences which appear more than expected in the file.

Overrepresented sequences

No overrepresented sequences

Warning

sequence is found to represent more than 0.1% of the total.

Failure

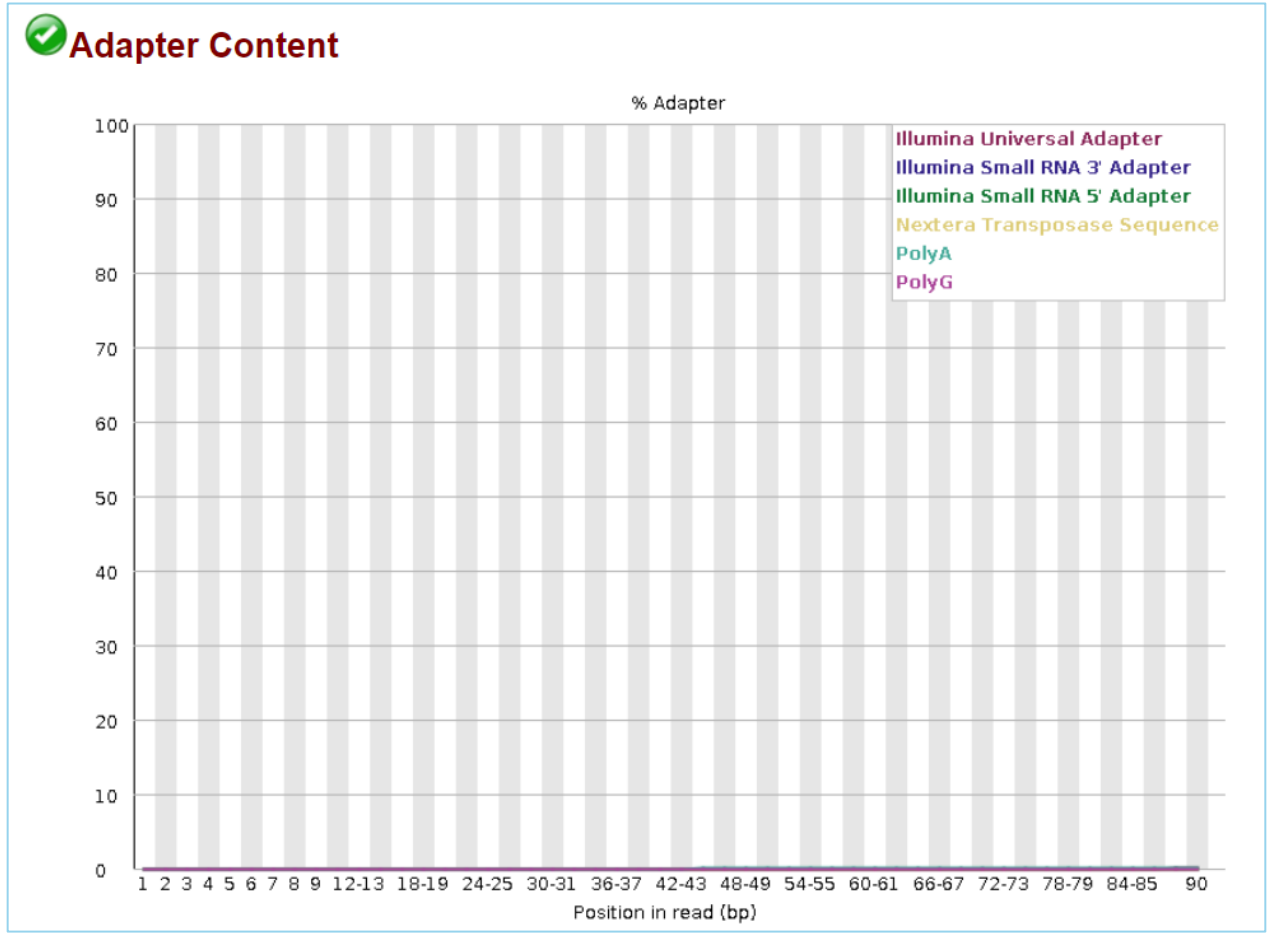
sequence is found to represent more than 1% of the total.

Overrepresented sequences

Sequence	Count	Percentage	Possible Source
CTGCTATGGCCACCAGACTCTCAGGCTCCATGCAGTGGCCAGCCTCATCG	2554	0.8349133703824779	No Hit
CAGCGGTCTAGTTTGAAGAACCTGACCCGAGTCTTGGTGACGAAGGCCAG	2463	0.8051650866296176	No Hit
GTTTGAAGAACCTGACCCGAGTCTTGGTGACGAAGGCCAGATTTGCGATC	1920	0.6276560967636483	No Hit
CCACAGGGTCCCAGGTCATGGGTACCGAGTCCAGGTCATAGTGCCGGATG	1219	0.39849624060150374	No Hit
GAAGAACCTGACCCGAGTCTTGGTGACGAAGGCCAGATTTGCGATCTTCA	1186	0.3877084014383786	No Hit
GGCAGGTGGACCCGGAGCCGCTGACAGAGGAGGTCAGCCCCTGAGTTGGA	1111	0.3631905851585486	No Hit
CACAGGGTCCCAGGTCATGGGTACCGAGTCCAGGTCATAGTGCCGGATGT	1079	0.35272965021248776	No Hit
GTCCCTGCTGGGGGCCAGGAGACGGTAGATGGAGCTGGGCAGGTGGACCC	1036	0.3386727688787185	No Hit

ADAPTER CONTENT

- a cumulative plot that shows the fraction of reads that contain a sequence library adapter at a given base position



Warning

Any sequence is present in more than 5% of all reads.

Failure

Any sequence is present in more than 10% of all reads.

PRE PROCESSING : TRIMMOMATIC

What Is Trimming in NGS?

1. Trimming refers to removing unwanted or low-quality sequences from high-throughput sequencing data.

2. Trimming helps to remove the regions of low confidence, sequencing artifacts, adapter sequences, and low-quality bases. This means that these artifacts and errors have to be removed, and this process of removal is known as trimming.

3. By performing trimming data, bioinformaticians can obtain cleaner, more accurate, and more reliable sequencing data. This is further important in obtaining high-quality results in various downstream bioinformatics applications, such as genome assembly, variant calling, gene expression analysis, and other biological investigations.

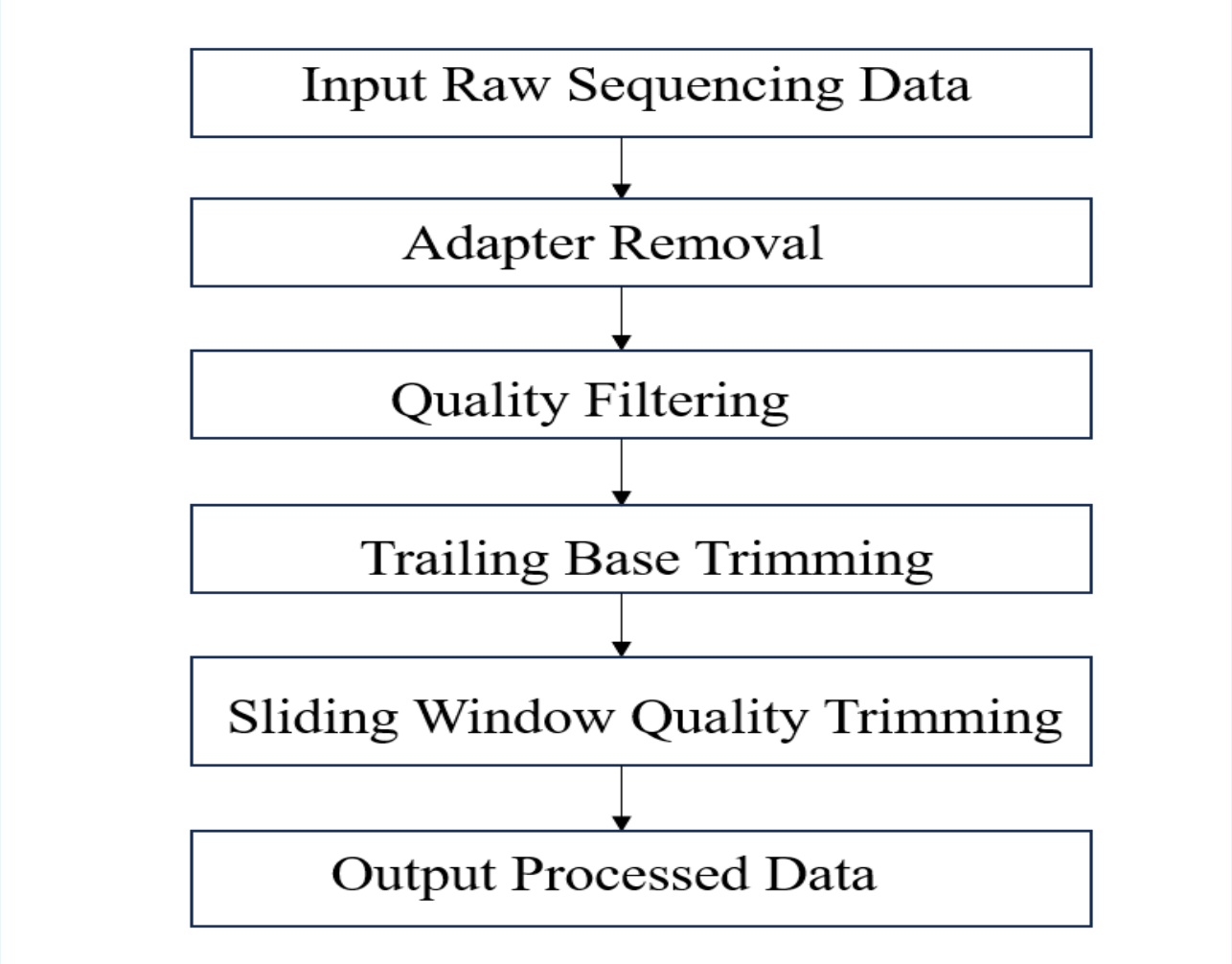
WHY IS TRIMMING DONE?

1. Trimming Is Necessary to Remove Adapters
2. Trimming Improves Overall Quality of NGS Data
3. In the Process of Trimming the Read Length Normalization Is Done
4. Trimming Also Helps in Removing Contaminants
5. Trimming of NGS Data Leads to Error Correction

What Is Trimmomatic?

1. Trimmomatic is one of the most popular bioinformatics tools for quality control (QC) and next-generation sequencing (NGS) data preprocessing.
2. It is widely used due to its efficiency, flexibility, and ability to work with various sequencing data formats.
3. Trimmomatic's main functionality is to remove low-quality regions and sequencing artifacts from raw NGS reads, ensuring that only high-quality, reliable data is used for the downstream analysis.
4. Trimmomatic is a command-line tool, and it is developed in Java.

STEPS:


Input Raw Sequencing Data
Adapter Removal
Quality Filtering
Trailing Base Trimming
Sliding Window Quality Trimming
Output Processed Data

Step 1: Input Raw Sequencing Data

Start with the input of raw sequencing data, typically in FASTQ format, which includes sequences and associated quality scores.

Step 2: Adapter Removal

Trimmomatic begins by identifying and removing adapter sequences from the raw reads. Adapter sequences are short DNA fragments used during the sequencing process that may still be present in the reads.

The tool uses two approaches to detect technical sequences within the reads: simple mode and palindrome mode

Simple mode works by finding an approximate match between the read and the user-supplied technical sequence, while palindrome mode looks for a palindromic match to detect adapter sequences in the reads

Step 3: Quality Filtering

Trimmomatic filters out low-quality reads based on the quality scores associated with each base position. Reads with low-quality scores are removed to improve the overall quality of the sequencing data. This step helps eliminate unreliable data that may adversely affect downstream analysis.

Step 4: Trailing Base Trimming

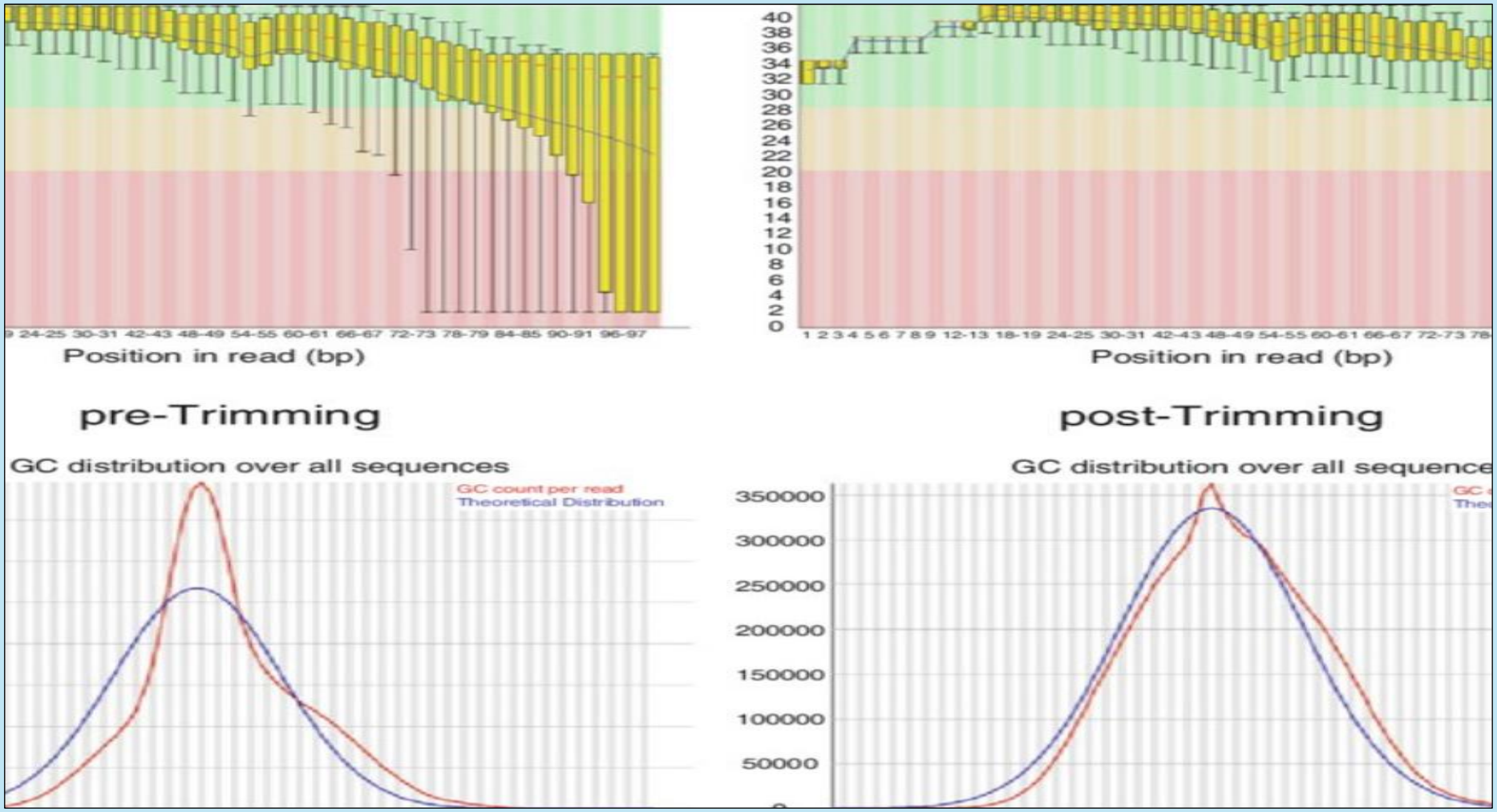
Trimmomatic trims bases from the 3' end (trailing end) of reads if their quality scores drop below a specified threshold. Trimming low-quality bases helps improve the accuracy of downstream analysis by removing unreliable data.

Step 5: Sliding Window Quality Trimming

Trimmomatic performs a sliding window analysis to identify regions of low quality within reads. If the average quality score within a window falls below a specified threshold, bases from the 3' end of the read are trimmed until the quality improves. This step helps remove regions of poor sequencing quality, ensuring the reliability of the sequencing data.

Step 6: Output Processed Data

The processed reads, with adapters removed and low-quality regions trimmed, are saved as output data. The output data is typically in FASTQ format and is ready for downstream analysis such as genome assembly, variant calling, or gene expression analysis.



Comparison between Pre-Trimming and Post-Trimming data

Steps after trimming process:

Quality Control Assessment: This step helps in ensuring that the trimming process did not introduce any biases or errors and that the data is of high quality for downstream analysis by using tools like FastQC

Alignment or Mapping: Align or map the trimmed reads to a reference genome using alignment tools such as Bowtie2, BWA, or HISAT2 for DNA sequencing or STAR for RNA sequencing.

Variant Calling :If you're working with DNA sequencing data, you may perform variant calling to identify genetic variations such as single nucleotide polymorphisms (SNPs) or small insertions/deletions (indels). Tools like GATK, FreeBayes, or SAMtools can be used for variant calling.

Gene Expression Analysis: For RNA sequencing data, if your goal is gene expression analysis, you may quantify gene expression levels using tools such as featureCounts, HTSeq, or Salmon.

Functional Analysis: After identifying differentially expressed genes or variants, you may perform functional analysis to understand the biological significance of your findings. This may involve pathway analysis, gene ontology enrichment analysis, or other functional annotation methods.

Visualization and Interpretation: Visualize the results using plots, graphs, or other visualization tools

Home | Download & Installation | Galaxy Usage | Command-line Usage | License | Useful Links | Contact

Here are screen-shots of the tool's pages in Galaxy.

- Galaxy Usage
- FASTA/Q Information tools
  - Quality Statistics
  - Quality Boxplot
  - Nucleotide Distribution
- FASTA/Q Manipulation Tools
  - FASTA/Q Clipper
  - FASTA/Q Trimmer
  - FASTA/Q End Trimmer
  - FASTQ Quality Trimmer
  - FASTA/Q Renamer
  - FASTA/Q Collapser
  - FASTA UnCollapser
  - UnCollapse rows (in a text file)
  - Artifacts Filter
  - FASTQ Quality Filter
  - FASTQ/A Reverse Complement

Home page of FASTX-Toolkit

FASTA/Q Information tools (screen-shots from Galaxy)

Quality Staistics

Quality Statistics

Library to analyse:

Execute

What it does

Creates quality statistics report for the given Solexa/FASTQ library.

TIP: This statistics report can be used as input for **Quality Score** and **Nucleotides Distribution** tools.

The output file will contain the following fields:

- column = column number (1 to 36 for a 36-cycles read solexa file)
- count = number of bases found in this column.
- min = Lowest quality score value found in this column.
- max = Highest quality score value found in this column.
- sum = Sum of quality score values for this column.
- mean = Mean quality score value for this column.
- Q1 = 1st quartile quality score.
- med = Median quality score.
- Q3 = 3rd quartile quality score.
- IQR = Inter-Quartile range (Q3-Q1).
- lW = 'Left-Whisker' value (for boxplotting).
- rW = 'Right-Whisker' value (for boxplotting).
- A_Count = Count of 'A' nucleotides found in this column.
- C_Count = Count of 'C' nucleotides found in this column.
- G_Count = Count of 'G' nucleotides found in this column.
- T_Count = Count of 'T' nucleotides found in this column.
- N_Count = Count of 'N' nucleotides found in this column.

Output Example:

column	count	min	max	sum	mean	Q1	med	Q3	IQR	lW	rW	A_Count	C_Count	G_Count	T_Count	N_Count
1	6362991	-4	40	250734117	39.41	40	40	40	0	40	40	1396976	1329101	678730	2958184	0
2	6362991	-5	40	250531036	39.37	40	40	40	0	40	40	1786786	1055766	1738025	1782414	0
3	6362991	-5	40	248722469	39.09	40	40	40	0	40	40	2296384	984875	1443989	1637743	0
4	6362991	-5	40	247654797	38.92	40	40	40	0	40	40	1683197	1410855	1722633	1546306	0
5	6362991	-4	40	248214827	39.01	40	40	40	0	40	40	2536861	1167423	1248968	1409739	0
6	6362991	-5	40	248499903	39.05	40	40	40	0	40	40	1598956	1236081	1568608	1959346	0
7	6362991	-4	40	247719760	38.93	40	40	40	0	40	40	1692667	1822140	1496741	1351443	0
8	6362991	-5	40	245745205	38.62	40	40	40	0	40	40	2230936	1343260	1529928	1258867	0
9	6362991	-5	40	245766735	38.62	40	40	40	0	40	40	1702064	1306257	1336511	2018159	0
10	6362991	-5	40	245089706	38.52	40	40	40	0	40	40	1519917	1446370	1450995	1945709	0
11	6362991	-5	40	242641359	38.13	40	40	40	0	40	40	1717434	1282975	1387804	1974778	0
12	6362991	-5	40	242026113	38.04	40	40	40	0	40	40	1662872	1202041	1519721	1978357	0
13	6362991	-5	40	238704245	37.51	40	40	40	0	40	40	1549965	1271411	1973291	1566681	1643
14	6362991	-5	40	235622401	37.03	40	40	40	0	40	40	2101301	1141451	1603990	1515774	475
15	6362991	-5	40	230766669	36.27	40	40	40	0	40	40	2344003	1058571	1440466	1519865	86
16	6362991	-5	40	224466237	35.28	38	40	40	2	35	40	2203515	1026017	1474060	1651582	7817
17	6362991	-5	40	219990002	34.57	34	40	40	6	25	40	1522515	1125455	2159183	1555765	73
18	6362991	-5	40	214104778	33.65	30	40	40	10	15	40	1479795	2068113	1558400	1249337	7346

Quality Boxplot

Quality Score

Statistics report file (output of 'FASTQ Statistics' tool):

Execute

What it does

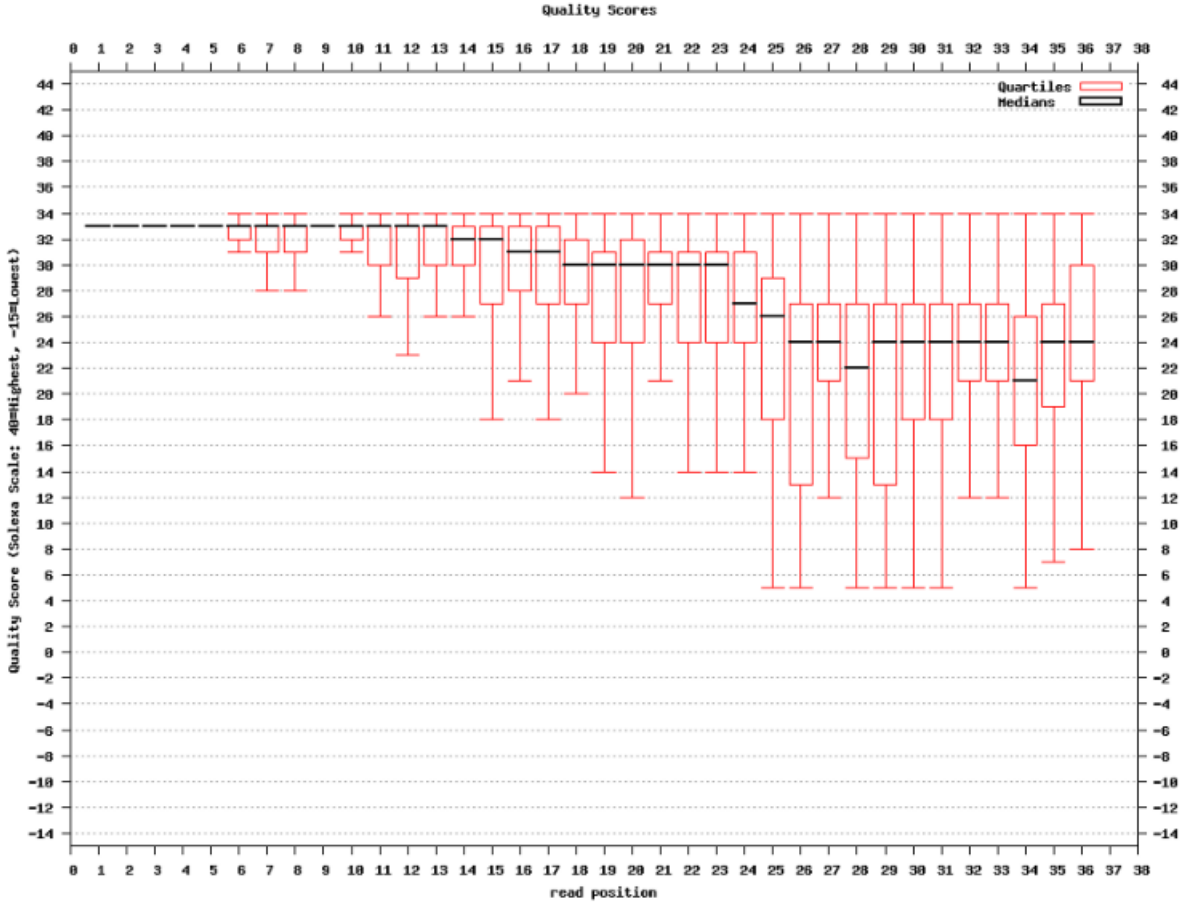
Creates a boxplot graph for the quality scores in the library.

TIP: Use the **FASTQ Statistics** tool to generate the report file needed for this tool.

Output Examples

- Black horizontal lines are medians
- Rectangular red boxes show the Inter-quartile Range (IQR) (top value is Q3, bottom value is Q1)
- Whiskers show outlier at max. 1.5*IQR

A relatively good quality library (median quality degrades towards later cycles):



Nucleotide Distribution

Nucleotides Distribution

Statistics Text File (output of 'FASTX Statistics' tool):

Execute

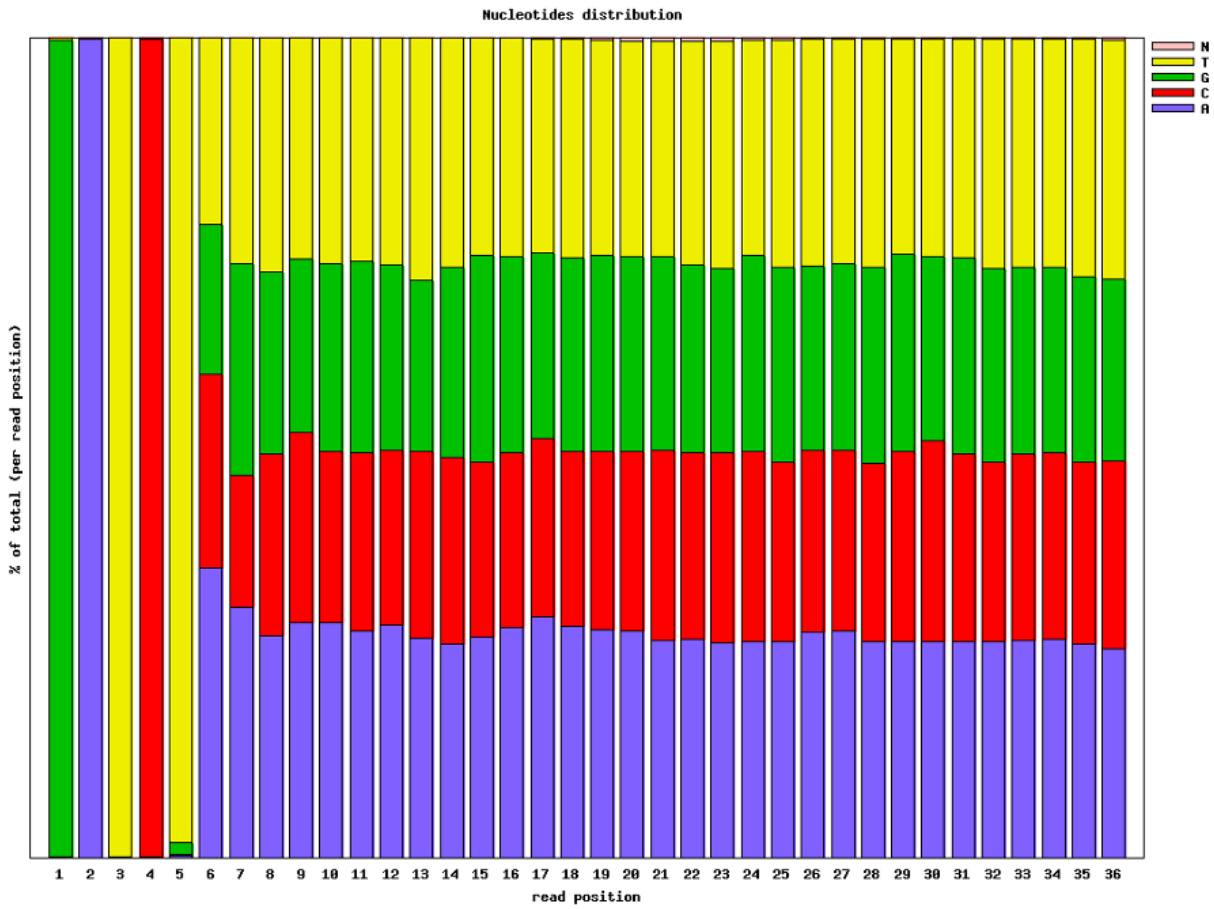
What it does

Creates a stacked-histogram graph for the nucleotide distribution in the Solexa library.

TIP: Use the **FASTQ Statistics** tool to generate the report file needed for this tool.

Output Examples

The following chart clearly shows the barcode used at the 5'-end of the library: **GATCT**



FASTA/Q Manipulation Tools (screen-shots from Galaxy)

FASTA/Q Clipper

Clipper

Library to clip:

Minimum sequence length (after clipping, sequences shorter than this length will be discarded):
15

Source:
Standard (select from the list below)

Choose Adapter:
Standard Illumina (CTGTAGGCACCATCAA

enter non-zero value to keep the adapter sequence and x bases that follow it:
0
use this for hairpin barcoding. keep at 0 unless you know what you're doing.

Discard sequences with unknown (N) bases:
Yes

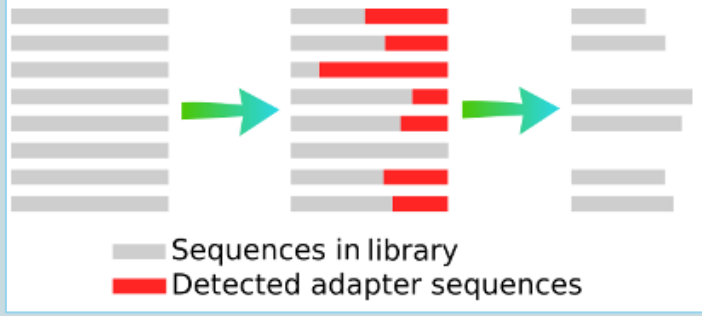
Output options:
Output only clipped seqeunces (i.e. sequen

Execute

What it does

This tool clips adapters from the 3'-end of the sequences in a FASTA/FASTQ file.

Clipping Illustration:



FASTA/Q Trimmer

Trim

Library to clip:

First base to keep:

1

Last base to keep:

21

Execute

What it does

This tool trims (cut bases from) sequences in a FASTA/Q file.

Example

Input Fasta file (with 36 bases in each sequences):

```
>1-1
TATGGTCAGAAACCATATGCAGAGCCTGTAGGCACC
>2-1
CAGCGAGGCTTTAATGCCATTTGGCTGTAGGCACCA
```

Trimming with First=1 and Last=21, we get a FASTA file with 21 bases in each sequences (starting from the first base):

```
>1-1
TATGGTCAGAAACCATATGCA
>2-1
CAGCGAGGCTTTAATGCCATT
```

Trimming with First=6 and Last=10, will generate a FASTA file with 5 bases (bases 6,7,8,9,10) in each sequences:

```
>1-1
TCAGA
>2-1
AGGCT
```

FASTA/Q Renamer

Rename

FASTQ/A Library to rename:

1: Pasted Entry

Rename sequence identifiers to:

Nucleotides sequence

Execute

What it does

This tool renames the sequence identifiers in a FASTQ/A file.

Use this tool at the beginning of your workflow, as a way to keep the original sequence (before trimming,clipping,barcode-removal, etc).

Example

The following Solexa-FASTQ file:

```
@CSHL_4_FC042GAMMII_2_1_517_596
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
+CSHL_4_FC042GAMMII_2_1_517_596
40 40 40 40 40 40 40 40 40 40 38 40 40 40 40 40 14 40 40 40 40 40 36 40 13 14 24 24 9 24 9 40 10 10 15 40
```

Renamed to **nucleotides sequence**:

```
@GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
+GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
40 40 40 40 40 40 40 40 40 40 38 40 40 40 40 40 14 40 40 40 40 40 36 40 13 14 24 24 9 24 9 40 10 10 15 40
```

Renamed to **numeric counter**:

```
@1
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
+1
40 40 40 40 40 40 40 40 40 40 38 40 40 40 40 40 14 40 40 40 40 40 36 40 13 14 24 24 9 24 9 40 10 10 15 40
```

FASTA Collapser

Collapse

Library to collapse:

Execute

What it does

This tool collapses identical sequences in a FASTA file into a single sequence.

Example

Example Input File (Sequence "ATAT" appears multiple times):

```
>CSHL_2_FC0042AGLLOO_1_1_605_414
TGCG
>CSHL_2_FC0042AGLLOO_1_1_537_759
ATAT
>CSHL_2_FC0042AGLLOO_1_1_774_520
TGGC
>CSHL_2_FC0042AGLLOO_1_1_742_502
ATAT
>CSHL_2_FC0042AGLLOO_1_1_781_514
TGAG
>CSHL_2_FC0042AGLLOO_1_1_757_487
TTCA
>CSHL_2_FC0042AGLLOO_1_1_903_769
ATAT
>CSHL_2_FC0042AGLLOO_1_1_724_499
ATAT
```

Example Output file:

```
>1-1
TGCG
>2-4
ATAT
>3-1
TGGC
>4-1
TGAG
>5-1
TTCA
```

Original Sequence Names / Lane descriptions (e.g. "CSHL_2_FC0042AGLLOO_1_1_742_502") are discarded.

The output seqeunce name is composed of two numbers: the first is the sequence's number, the second is the multiplicity value.

The following output:

```
>2-4
ATAT
```

means that the sequence "ATAT" is the second sequence in the file, and it appeared 4 times in the input FASTA file.

FASTA UnCollapser

Uncollapse

Collapsed FASTA file:

16: FASTA File

Execute

What it does

This tool uncollapses a previously-collapsed FASTA file. It reads each collapsed sequence and generates multiple sequences based on the collapsed read count.

Example

Example Input - a collapsed FASTA file (Sequence "ATAT" has four collapsed reads):

```
>1-1
TGCG
>2-4
ATAT
```

Example Output - uncollapsed FASTA file (Sequence "ATAT" now appears as 4 separate sequences):

```
>1
TGCG
>2
ATAT
>3
ATAT
>4
ATAT
>5
ATAT
```

The original sequence id (with the read counts) are discarded, with the sequence given a numerical name.

This tool is based on FASTX-toolkit by Assaf Gordon.

UnCollapse row (in a text file)

Uncollapse rows

Library to uncollapse:

17: Pasted Entry

Column with collased sequence-identifier:

c1

This column contains the sequence id from a collapsed FASTA file in the form of "(seq number)-(read count)" (e.g. 15-4). Use 10 if you're analyzing BLAT output

Execute

What it does

This tool reads a row (in a table) containing a collapsed sequence ID, and duplicates the .

You must specify the column containing the collapsed sequence ID (e.g. 15-4).

Example Input File

The following input file contains two collapsed sequence identifiers at column 10: *84-2* and *87-5*

(meaning the first has multiplicity-count of 2 and the second has multiplicity count of 5):

```
23    0    0    0    0    0    0    0    +    84-2 ...
22    0    0    0    0    0    0    0    +    87-5 ...
```

Output Example

After **uncollapsing** (on column 10), the line of the first sequence-identifier is repeated *twice*, and the line of the second sequence-identifier is repeated *five* times:

```
23    0    0    0    0    0    0    0    +    84-2 ...
23    0    0    0    0    0    0    0    +    84-2 ...
22    0    0    0    0    0    0    0    +    87-5 ...
22    0    0    0    0    0    0    0    +    87-5 ...
22    0    0    0    0    0    0    0    +    87-5 ...
22    0    0    0    0    0    0    0    +    87-5 ...
22    0    0    0    0    0    0    0    +    87-5 ...
```

Uncollapsing a text file allows analsys of collapsed FASTA files to be used with any tool which doesn't 'understand' collapsed multiplicity counts.

See the *Collapse* tool in the *FASTA Manipulation* category for more details about collapsing FASTA files.

This tool is based on FASTX-toolkit by Assaf Gordon.

Artifacts Filter

Artifacts Filter

Library to filter:

Execute

What it does

This tool filters sequencing artifacts (reads with all but 3 identical bases).

The following is an example of sequences which will be filtered out:

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAACACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAACACAAAAAAAAAAAAAAAAAAAAAAAAAACACAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
AAAAACACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

FASTQ/A Reverse Complement

Reverse-Complement

Library to reverse-complement:

Execute

What it does

This tool reverse-complements each sequence in a library. If the library is a FASTQ, the quality-scores are also reversed.

Example

Input FASTQ file:

```
@CSHL_1_FC42AGWWWXX:8:1:3:740
TGTCTGTAGCCTCNTCCTTGTAATTCAAAGNNGGTA
+CSHL_1_FC42AGWWWXX:8:1:3:740
33 33 33 34 33 33 33 33 33 33 33 33 27 5 27 33 33 33 33 33 33 27 21 27 33 32 31 29 26 24 5 5 15 17 27 26
```

Output FASTQ file:

```
@CSHL_1_FC42AGWWWXX:8:1:3:740
TACCNNCTTTGAATTACAAGGANGAGGCTACAGACA
+CSHL_1_FC42AGWWWXX:8:1:3:740
26 27 17 15 5 5 24 26 29 31 32 33 27 21 27 33 33 33 33 33 33 27 5 27 33 33 33 33 33 33 33 33 34 33 33 33
```

FASTQ-to-FASTA converter

FASTQ to FASTA

FASTQ Library to convert:

Discard sequences with unknown (N) bases :

yes

Rename sequence names in output file (reduces file size):

yes

Execute

What it does

This tool converts data from Solexa format to FASTA format (scroll down for format description).

Example

The following data in Solexa-FASTQ format:

```
@CSHL_4_FC042GAMMII_2_1_517_596
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
+CSHL_4_FC042GAMMII_2_1_517_596
40 40 40 40 40 40 40 40 40 40 38 40 40 40 40 40 14 40 40 40 40 40 36 40 13 14 24 24 9 24 9 40 10 10 15 40
```

Will be converted to FASTA (with 'rename sequence names' = NO):

```
>CSHL_4_FC042GAMMII_2_1_517_596
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
```

Will be converted to FASTA (with 'rename sequence names' = YES):

```
>1
GGTCAATGATGAGTTGGCACTGTAGGCACCATCAAT
```

FASTA Formatter

FASTA Width

Library to re-format:

10: [Pasted Entry] (reformatted)

New width for nucleotides strings:

0

Use 0 for single line outout.

Execute

What it does

This tool re-formats a FASTA file, changing the width of the nucleotides lines.

TIP: Outputting a single line (with **width** = **0**) can be useful for scripting (with **grep**, **awk**, and **perl**). Every odd line is a sequence identifier, and every even line is a nucleotides line.

Example

Input FASTA file (each nucleotides line is 50 characters long):

```
>Scaffold3648
AGGAATGATGACTACAATGATCAACTTAACCTATCTATTTAATTTAGTTC
CCTAATGTCAGGGACCTACCTGTTTTTGTTATGTTTGGGTTTTGTTGTTG
TTGTTTTTTTAATCTGAAGGTATTGTGCATTATATGACCTGTAATACACA
ATTAAAGTCAATTTTAATGAACATGTAGTAAAAACT
>Scaffold9299
CAGCATCTACATAATATGATCGCTATTAAACTTAAATCTCCTTGACGGAG
TCTTCGGTCATAACACAAACCCAGACCTACGTATATGACAAAGCTAATAG
aactggtctttacctTTAAGTTG
```

Output FASTA file (with width=80):

```
>Scaffold3648
AGGAATGATGACTACAATGATCAACTTAACCTATCTATTTAATTTAGTTCCCTAATGTCAGGGACCTACCTGTTTTTGTT
ATGTTTGGGTTTTGTTGTTGTTGTTTTTTTAATCTGAAGGTATTGTGCATTATATGACCTGTAATACACAATTAAAGTCA
ATTTTAATGAACATGTAGTAAAAACT
>Scaffold9299
CAGCATCTACATAATATGATCGCTATTAAACTTAAATCTCCTTGACGGAGTCTTCGGTCATAACACAAACCCAGACCTAC
GTATATGACAAAGCTAATAGaactggtctttacctTTAAGTTG
```

Output FASTA file (with width=0 => single line):

```
>Scaffold3648
AGGAATGATGACTACAATGATCAACTTAACCTATCTATTTAATTTAGTTCCCTAATGTCAGGGACCTACCTGTTTTTGTTATGTTTGGGTTTTGTTGTTGTTGTTTTTTTAATCTGAAGGTATTGTGCATT
>Scaffold9299
CAGCATCTACATAATATGATCGCTATTAAACTTAAATCTCCTTGACGGAGTCTTCGGTCATAACACAAACCCAGACCTACGTATATGACAAAGCTAATAGaactggtctttacctTTAAGTTG
```

FASTQ/A barcode splitter

Barcode Splitter

Barcodes to use:

Library to split:

Barcodes found at:

Start of sequence (5' end)

Number of allowed mismatches:

2

Number of allowed barcodes nucleotide deletions:

0

Execute

What it does

This tool splits a solexa library (FASTQ file) or a regular FASTA file to several files, using barcodes as the split criteria.

Barcode file Format

Barcode files are simple text files. Each line should contain an identifier (descriptive name for the barcode), and the barcode itself (A/C/G/T), separated by a TAB character. Example:

```
#This line is a comment (starts with a 'number' sign)
BC1 GATCT
BC2 ATCGT
BC3 GTGAT
BC4 TGTCT
```

For each barcode, a new FASTQ file will be created (with the barcode's identifier as part of the file name). Sequences matching the barcode will be stored in the appropriate file.

One additional FASTQ file will be created (the 'unmatched' file), where sequences not matching any barcode will be stored.

The output of this tool is an HTML file, displaying the split counts and the file locations.

Output Example

Barcode	Count	Location
BC1	69006	http://tango/barcode_splits/2008-08-14_2328__small__BC1.txt
BC2	114576	http://tango/barcode_splits/2008-08-14_2328__small__BC2.txt
BC3	7	http://tango/barcode_splits/2008-08-14_2328__small__BC3.txt
BC4	64948	http://tango/barcode_splits/2008-08-14_2328__small__BC4.txt
unmatched	1463	http://tango/barcode_splits/2008-08-14_2328__small__unmatched.txt
total	250000	

References:

- Shi, H., Xu, X., School of Computer and Information Engineering, Beijing University of Agriculture, China, & Communication Technology Bureau, Xinhua News Agency, China. (2016). Learning the Sequences Quality Control of Bioinformatics Analysis Method. In International Conference on Education, E-learning and Management Technology (EEMT 2016). https://www.atlantis-press.com/article/25860057.pdf
- Shi, H., Li, W., Xu, X., School of Computer and Information Engineering, Beijing University of Agriculture, China, College of Plant Science and Technology, Beijing University of Agriculture, China, & Communication Technology Bureau, Xinhua News Agency, China. (2016). Learning the comparing and converting method of sequence PhRed quality score. In 6th International Conference on Management, Education, Information and Control (MEICI 2016). https://www.atlantis-press.com/article/25863649.pdf
- Babraham Bioinformatics - FastQC A Quality Control tool for High Throughput Sequence Data. (n.d.). https://www.bioinformatics.babraham.ac.uk/projects/fastqc/
- Index of /projects/fastqc/Help. (n.d.). https://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/
- Sanger, J. M. a. V. C. (n.d.). The FASTQ format and quality control. https://slideplayer.com/slide/13772534/
- Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics (Oxford, England)*, *30*(15), 2114–2120. https://doi.org/10.1093/bioinformatics/btu170
- Hamzić, E. (2024, January 3). All You Need To Know About Trimmomatic & NGS Data Trimming. BioComputiX. https://www.biocomputix.com/post/trimming-ngs-data-trimmomatic
- FASTX-Toolkit. (n.d.). http://hannonlab.cshl.edu/fastx_toolkit/

THANK YOU